

# Documentación de pruebas unitarias

Mariana Méndez Morales

ADSO 2696224

Instructor German Alberto Angarita Henao

Prueba #1: En esta prueba se hace la suma de dos números enteros

```
Prueba 1 > JS sumaEnteros.js > sumaEnteros
1  function sumaEnteros(numero) {
2      let suma = 0;
3      for (let i = 1; i <= numero; i++) {
4          suma += i;
5      }
6      return suma;
7  }
8
9  module.exports = sumaEnteros;
```

```
Prueba 1 > JS sumaEnteros.test.js > test('Calcula la suma de enteros hasta un número dado') callback
1  const sumaEnteros = require('./sumaEnteros');
2
3  test('Calcula la suma de enteros hasta un número dado', () => {
4      expect(sumaEnteros(4)).toBe(10);
5      expect(sumaEnteros(6)).toBe(21);
6  });
```

Prueba #2: En esta prueba se verifica si una persona es mayor o menor de edad

Prueba 2 > JS edad.js > ...

```
1  function edad(edad) {
2      edad = 18;
3      if (edad >= 18) {
4          console.log("Eres mayor ");
5      } else {
6          console.log("Eres mayor ");
7      }
8      return edad;
9  }
10 module.exports = edad;
```

Prueba 2 > JS edad.test.js > ...


```
1  const edad = require('./edad');
2
3  test('Si edad es = 18 decir "Eres mayor" sino decir "Eres menor"', () => {
4      expect(edad(edad)).toBe(18);
5  });
```

Prueba #3: En esta prueba se calcula el promedio de edades de un grupo de mascotas que están almacenadas en un array

```
Prueba 3 > JS mascota.js > mascota
1  function mascota(mascotas) {
2      if (mascotas.length === 0) {
3          return 0;
4      }
5
6      let sumaEdades = 0;
7      for (const mascota of mascotas) {
8          sumaEdades += mascota.edad;
9      }
10     const promedio = sumaEdades / mascotas.length;
11
12     return promedio;
13 }
14 module.exports = mascota;
15
16
```

```
Prueba 3 > JS mascota.test.js > ...
1  const mascota = require('./mascota');
2
3  test('Calcular la edad promedio de las mascotas', () => {
4      const mascotas = [
5          { nombre: 'Luna', edad: 5 },
6          { nombre: 'Max', edad: 3 },
7          { nombre: 'Coco', edad: 7 },
8      ];
9
10     expect(mascota(mascotas)).toBe(5);
11 });
12
13 test('Calcular la edad promedio de un grupo vacío de mascotas', () => {
14     const mascotas = [];
15     expect(mascota(mascotas)).toBe(0);
16 });
```

Prueba #4: En esta prueba se obtiene la edad máxima y la edad mínima de un grupo de estudiantes

Prueba 4 > JS estudiante.js >  estudiante

```
1  function estudiante(estudiantes) {
2      if (estudiantes.length === 0) {
3          return false;
4      }
5
6      let edadMinima = estudiantes[0].edad;
7      let edadMaxima = estudiantes[0].edad;
8
9      for (const estudiante of estudiantes) {
10         if (estudiante.edad < edadMinima) {
11             edadMinima = estudiante.edad;
12         }
13         if (estudiante.edad > edadMaxima) {
14             edadMaxima = estudiante.edad;
15         }
16     }
17
18     return {
19         minima: edadMinima,
20         maxima: edadMaxima,
21     };
22 }
23
24 module.exports = estudiante;
```

Prueba 4 > JS estudiante.test.js > ...

```
1  const estudiante = require('./estudiante');
2
3  test('Obtiene edad mínima y máxima de los estudiantes', () => {
4      const estudiantes = [
5          { nombre: 'Mari', edad: 17 },
6          { nombre: 'Deivid', edad: 20 },
7          { nombre: 'Pablo', edad: 18 },
8      ];
9
10     expect(estudiante(estudiantes)).toEqual({
11         minima: 17,
12         maxima: 20,
13     });
14 });
```

Prueba #5: En esta prueba verifica si una cadena de texto es una palíndromo (una palabra que se lee de igual manera de izquierda a derecha tanto como de derecha a izquierda “somos”)

```
Prueba 5 > JS invertida.js > ...
1 function invertida(cadena) {
2     const cadenaLimpia = cadena.toLowerCase().replace(/^[^a-zA-Z0-9]/g, '');
3     const cadenaInvertida = cadenaLimpia.split('').reverse().join('');
4     return cadenaLimpia === cadenaInvertida;
5 }
6
7 module.exports = invertida;
8
9
```

```
Prueba 5 > JS invertida.test.js > ...
1 const invertida = require('./invertida');
2
3 test('Verifica si una cadena es un palíndromo', () => {
4     expect(invertida('anilina')).toBe(true);
5     expect(invertida('reconocer')).toBe(true);
6     expect(invertida('Hola, mundo')).toBe(false);
7     expect(invertida('soMos')).toBe(true);
8 });
```

Resultado de pruebas:

```
> jest
PASS Prueba 5/invertida.test.js
PASS Prueba 4/estudiante.test.js
PASS Prueba 3/mascota.test.js
PASS Prueba 2/edad.test.js
  ● Console

    console.log
      Eres mayor

      at log (D:\German A\Mari Morita (Nodejs-Json)...\Prueba 2/edad.js:4:17)

PASS Prueba Ejemplo/suma.test.js
PASS Prueba 1/sumaEnteros.test.js

Test Suites: 6 passed, 6 total
Tests:       7 passed, 7 total
Snapshots:   0 total
Time:        9.164 s
Ran all test suites.
PS D:\German A\Mari Morita (Nodejs-Json)>
```